



# ***Защита в операционных системах***

Вязанкин Олег Николаевич

к.ф.-м.н., доцент кафедры ПОЗИ, ИМИТ

[ovyazankin@gmail.com](mailto:ovyazankin@gmail.com)



## Тема 5. Аудит и обнаружение вторжений.



# План

- Общие сведения
- Системы обнаружения вторжений
- Аудит в Windows
- Аудит в UNIX



# Общие сведения

Процедура аудита применительно к защищенным компьютерным системам заключается в регистрации в специальном журнале, называемом *журналом аудита* или *журналом безопасности*, событий, которые могут представлять опасность для системы.

Пользователи системы, обладающие правом чтения этого журнала, называются *аудиторами*.

# Общие сведения

Необходимость включения в защищенную систему функций аудита диктуется следующими обстоятельствами.

- Подсистема защиты компьютерной системы, не обладая интеллектом, неспособна отличить случайные ошибки пользователей от злонамеренных действий.
- Администраторы защищаемой системы должны иметь возможность получать информацию не только о текущем состоянии системы, но и о том, как она функционировала в недавнем прошлом. Журнал аудита дает такую возможность, накапливая информацию о важных событиях, связанных с безопасностью операционной системы.
- Если администратор обнаружил, что против защищаемой системы проведена успешная атака, ему важно выяснить, когда она была начата и каким образом она осуществлялась. При наличии в системе подсистемы аудита не исключено, что вся необходимая информация содержится в журнале аудита.

# Общие сведения

Некоторые эксперты по компьютерной безопасности полагают, что привилегия работать с подсистемой аудита не должна предоставляться администраторам операционной системы. Другими словами, множества администраторов и множество аудиторов не должны пересекаться.

При этом создается ситуация, когда администратор не может выполнять несанкционированные действия без того, чтобы это тут же стало известно аудиторам, что повышает защищенность системы от несанкционированных действий администраторов.

Однако на практике отделение аудиторов от администраторов применяется редко, что обусловлено следующими причинами:

- обслуживание аудита на одном компьютере занимает существенно меньше времени, чем администрирование одного компьютера. Если организация невелика и в ней предусмотрены всего 1-2 штатные должности системных администраторов, создавать отдельную штатную должность аудитора нецелесообразно;

- обслуживание аудита на одном компьютере занимает существенно меньше времени, чем администрирование одного компьютера. Если организация невелика и в ней предусмотрены всего 1-2 штатные должности системных администраторов, создавать отдельную штатную должность аудитора нецелесообразно;

# Общие сведения

- администраторы и аудиторы часто вступают в приятельские отношения, в результате чего аудитор далеко не всегда докладывает начальнику об обнаруженных злоупотреблениях администраторов. Бывает, что аудиторы не только прикрывают злоупотребления администраторов, но и сами участвуют в них. Если отделение аудиторов от администраторов не подкреплено организационно-административными мерами, оно может оставаться чисто формальным — аудитор знает пароль администратора, администратор знает пароль аудитора, и оба они пользуются полномочиями друг друга по мере необходимости;
- при наличии в организации выделенного аудитора, недостаточно загруженного работой, аудитор часто вводит чрезмерно строгий контроль за действиями пользователей, негативно сказывающийся на моральном климате в коллективе. Практика показывает, что выделенные аудиторы более склонны к реализации «параноической» политики безопасности, чем аудиторы, совмещающие свои обязанности с обязанностями системного администратора.

Обычно отделение аудиторов от администраторов практикуется только в крупных организациях (корпорациях, банках и т. п.). При этом обязанности аудитора часто берет на себя должностное лицо, которому непосредственно подчиняются системные администраторы, либо его заместитель.

# Общие сведения

## Требования к аудиту

Подсистема аудита операционной системы должна удовлетворять следующим требованиям.

- Добавлять записи в журнал аудита могут только псевдопользователи, от имени которых выполняются системные процессы. Если предоставить эту возможность какому-то физическому пользователю, данный пользователь получит возможность компрометировать других пользователей, добавляя в журнал аудита соответствующие записи.
- Ни один субъект доступа, в том числе и сама операционная система, не имеет возможности редактировать или удалять отдельные записи в журнале аудита.
- Только пользователи-аудиторы, обладающие соответствующей привилегией, могут просматривать журнал аудита.
- Только пользователи-аудиторы могут очищать журнал аудита. После очистки журнала в него автоматически вносится запись о том, что журнал аудита был очищен, с указанием времени очистки журнала и имени пользователя, очистившего журнал. Система аудита должна поддерживать возможность сохранения журнала аудита перед очисткой в другом файле.





# Общие сведения

## Требования к аудиту

Для ограничения доступа пользователей к журналу аудита не всегда достаточно обычных средств разграничения доступа.

В подавляющем большинстве систем администратор локальной или корпоративной сети, используя свои привилегии, может прочитать и изменить содержимое любого файла, хранящегося на любом компьютере сети.

Поэтому, если принятая в системе политика безопасности предусматривает разделение администраторов и аудиторов, то для ограничения доступа к журналу аудита желательно применять дополнительные средства защиты, например, криптографические.

# Общие сведения

## Политика аудита

*Политика аудита* — это совокупность правил, определяющая то, какие события должны регистрироваться в журнале аудита.

Для обеспечения надежной защиты операционной системы в журнале аудита должны обязательно регистрироваться следующие события:

- попытки входа/выхода пользователей из системы;
- попытки изменения списка пользователей;
- попытки изменения политики безопасности, в том числе и политики аудита.

При определении политики аудита не следует ограничиваться регистрацией событий только из перечисленных классов. Окончательный выбор того, какие события должны регистрироваться в журнале аудита, возлагается на самих аудиторов. При этом политика аудита в значительной степени определяется спецификой информации, хранимой и обрабатываемой в операционной системе, и не зная этой специфики, давать какие-либо рекомендации бессмысленно.

# Общие сведения

## Политика аудита

- При выборе оптимальной политики аудита следует учитывать ожидаемую скорость заполнения журнала аудита.
- Политику аудита не следует рассматривать как нечто неизменное, заданное раз и навсегда. Политика аудита должна оперативно реагировать на изменения в конфигурации операционной системы, в характере хранимой и обрабатываемой информации, и, особенно, на выявленные попытки атаковать защищаемую операционную систему.
- В целом политика аудита является своего рода искусством, и выбор оптимальной политики в значительной мере определяется опытом и интуицией аудитора.



# Общие сведения

## Интерактивное оповещение аудитора

В некоторых конфигурациях операционных систем подсистема аудита помимо записи информации о зарегистрированных событиях в специальный журнал предусматривает возможность интерактивного оповещения аудиторов об этих событиях.

Когда аудитор начинает работу с операционной системой одного из компьютеров сети, операционные системы других компьютеров получают соответствующие сообщения, после чего при каждой регистрации события в журнале аудита одного из компьютеров копия информации об этом событии передается на терминал, с которым работает аудитор.

Как правило, для реализации подобного механизма требуется установка дополнительного программного обеспечения.



# Общие сведения

## Интерактивное оповещение аудитора

Множество событий, регистрируемых в журнале аудита, не обязательно должен совпадать с множеством событий, информация о которых передается аудиторам интерактивно.

Целесообразно так организовать интерактивное оповещение аудиторов, чтобы аудиторы получали оповещение только о наиболее важных событиях — в противном случае аудиторам будет трудно выделить в сплошном потоке сообщений по-настоящему полезную информацию.

Данная дополнительная функция подсистемы аудита позволяет аудиторам более оперативно реагировать на попытки преодоления злоумышленниками защиты операционной системы и тем самым повышает общую защищенность системы.

# Системы обнаружения вторжений

Логическим развитием концепции аудита является *система обнаружения вторжений* (COB), или *intrusion detection system* (IDS) — специализированное программное или программно-аппаратное средство, предназначенное для выявления успешных и неуспешных попыток осуществления несанкционированного доступа к ресурсам компьютерной системы или сети.

Системы обнаружения вторжений обладают двумя характеристическими отличиями от обычных систем аудита:

- система обнаружения вторжений не просто регистрирует отдельные события, происходящие в системе, но и анализирует их в совокупности, пытаясь обнаружить в последовательности зафиксированных событий признаки атаки;
- система обнаружения вторжений может оперативно реагировать на обнаруженные атаки, самостоятельно блокируя соответствующие функции системы до того, как нарушитель успел им воспользоваться.

До середины 2000-х годов системы обнаружения вторжений применялись на практике крайне редко. Однако современные тенденции в области информационной безопасности таковы, что область применения систем обнаружения вторжений становится все шире и системы обнаружения вторжений постепенно перестают быть «экзотикой».

# Системы обнаружения вторжений

Типичная архитектура системы обнаружения вторжений включает в себя следующие основные элементы:

- *сенсоры*, обеспечивающие сбор информации для последующего анализа;
- *анализаторы*, осуществляющие анализ полученной сенсорами информации;
- *хранилище* (как правило, базу данных), в которое помещаются результаты работы анализатора;
- *консоль управления*, обеспечивающую взаимодействие администратора безопасности с системой обнаружения вторжений.

По набору используемых сенсоров (способу сбора информации) системы обнаружения вторжений классифицируются на:

- *узловые*, или *хостовые* (host IDS, HIDS) — берут информацию от подсистемы аудита защищаемой операционной системы или системы управления базами данных, а также дополнительных защитных подсистем (контроля целостности, антивирусного мониторинга и т. п.);
- *сетевые* (network IDS, NIDS) — анализируют сетевой трафик;
- гибридные или смешанные.

По способности предпринимать активные действия в ответ на выявленные угрозы безопасности системы обнаружения вторжений классифицируются на:

- *активные* (*системы предотвращения вторжений, intrusion prevention systems, IPS*);
- *пассивные*.

# Системы обнаружения вторжений

## Сенсоры

Сенсоры HIDS делятся на пять основных типов:

- сенсоры журналов;
- сенсоры признаков;
- сенсоры системных вызовов;
- сенсоры поведения приложений;
- сенсоры целостности файлов.

Сенсоры NIDS представляют собой программные или программно-аппаратные sniffеры (анализаторы трафика), осуществляющие перехват сетевого трафика одного компьютера или целого сегмента локальной сети.



# Системы обнаружения вторжений

## Анализаторы

Большинство IDS могут анализировать не только информацию, полученную непосредственно с сенсоров в реальном времени, но и работать с журналами, содержащими ранее собранную информацию. Это позволяет проводить при необходимости повторный «разбор полетов» для различных инцидентов, связанных с информационной безопасностью, например, проверять, способна ли перенастроенная система обнаружения вторжений обнаружить атаку, ранее прошедшую незамеченной.

Анализаторы IDS анализируют собранную сенсорами информацию на предмет сходства с типичными атаками нарушителей. Современные системы обнаружения вторжений довольно надежно детектируют сканирование портов, с которого начинается большинство сетевых атак, а также попытки программных закладок, проникших внутрь защищаемой сети, связываться со своими хозяевами через Интернет.

При этом, в отличие от традиционных систем разграничения доступа и аудита, системы обнаружения вторжений реагируют не на каждое зафиксированное событие в отдельности, а на всю последовательность событий в совокупности.



# Системы обнаружения вторжений

## Анализаторы

В некоторых системах обнаружения вторжений используется многоуровневая модель построения анализаторов, что позволяет системе обнаружения вторжений эффективно обнаруживать угрозы безопасности, затрагивающие сразу несколько компонент защищаемой системы.

Правила, реализуемые анализаторами IDS, бывают двух видов:

- сигнатурные;
- эвристические.

# Системы обнаружения вторжений

## Сигнатурные анализаторы

Сигнатурный анализатор ищет в анализируемом потоке информации так называемые *сигнатуры* или *сценарии атак* — четкие и недвусмысленные признаки определенных атак.

Примерами сигнатур могут служить:

- сигнатура подбора пароля — несколько неудачных попыток аутентификации с одного рабочего места;
- сигнатура сканирования портов — несколько попыток открытия различных портов защищаемого сервера одним и тем же клиентом;
- сигнатура эксплуатации уязвимости программного обеспечения — получение одного или нескольких определенных сетевых пакетов, пришедших на определенный порт.

# Системы обнаружения вторжений

## Сигнатурные анализаторы

Достоинства и недостатки:

- относительная простота реализации;
- более редкие ложные срабатывания;
- реагирует лишь на атаки, сигнатуры которых присутствуют в базе сигнатур;
- постоянная поддержка базы сигнатур в актуальном состоянии.

# Системы обнаружения вторжений

## Эвристические анализаторы

Эвристические анализаторы работают на основе эвристических правил, позволяющих выявлять аномальную активность в защищаемой системе.

Обычно в ходе функционирования системы активность отдельных ее компонент примерно одинакова и слабо меняется со временем. Если какая-то характеристика какой-то компоненты системы резко изменилась, это воспринимается как сигнал тревоги, например:

- если некий процесс начал потреблять заметно больше аппаратных ресурсов компьютера, чем раньше, возможно, в адресное пространство этого процесса внедрилась программная закладка;

- если некий компьютер генерирует необычно много исходящего SMTP-трафика, возможно, операционная система данного компьютера поражена сетевым вирусом, осуществляющим рассылки спам-почты.

# Системы обнаружения вторжений

## Эвристические анализаторы

Основные достоинства и недостатки:

- основным достоинством эвристических анализаторов является их способность более-менее адекватно реагировать на ранее неизвестные атаки злоумышленников;

- основным недостатком эвристических анализаторов является их склонность генерировать ложные тревоги.

Эвристический анализатор системы обнаружения вторжений в общем случае не гарантирует обнаружение атаки. Большинство атак могут быть так модифицированы, что их реализация не будет приводить к заметным всплескам активности тех или иных компонент атакуемой системы. Однако модифицированные атаки, как правило, менее эффективны, чем в оригинальном исполнении.

# Системы обнаружения вторжений

В настоящее время сигнатурные и эвристические анализаторы часто используются в совокупности. В будущем, с ростом «интеллектуальности» систем обнаружения вторжений, эвристические анализаторы, вероятно, станут основным видом анализаторов, применяемых в системах обнаружения вторжений.

Пока же в практической работе основная нагрузка ложится на сигнатурные анализаторы, а эвристические используются главным образом в пассивном режиме, когда решение о том, была ли выявлена атака или имела место ложная тревога, принимает человек.

Сопровождение системы обнаружения вторжений требует от администраторов защищаемой системы довольно больших усилий и высокой квалификации. Малоквалифицированный администратор, обслуживающий систему обнаружения вторжений, как правило, постепенно отключает функции системы, смысл которых ему неясен, и со временем IDS приходит в состояние, когда анализатор игнорирует большую часть тревожных сигналов, поступающих от сенсоров. Такая система обнаружения вторжений не приносит никакой пользы и даже, напротив, приносит вред, поскольку создает у пользователей и администраторов системы ложное чувство защищенности.

# Системы обнаружения вторжений

Несмотря на свою высокую эффективность, системы обнаружения вторжений не являются панацеей, гарантированно пресекающей все атаки злоумышленников. Это обусловлено следующими факторами.

- Система обнаружения вторжений не обладает полноценным интеллектом. Любая система обнаружения вторжений может быть обманута достаточно квалифицированным и удачливым злоумышленником.

- Система обнаружения вторжений нуждается в постоянном сопровождении администратора безопасности. Журналы, создаваемые системой обнаружения вторжений, должны регулярно просматриваться и анализироваться человеком, обладающим, в отличие от системы обнаружения вторжений, полноценным интеллектом.

- Возможности системы обнаружения вторжений по автоматической реакции на зафиксированные угрозы весьма ограничены.

- 

- Ни одна система обнаружения вторжений не способна обнаруживать абсолютно любые атаки, для любой системы обнаружения вторжений существуют области защищаемой системы, находящиеся вне пределов видимости ее сенсоров.

- Как и любой программный продукт, система обнаружения вторжений может иметь уязвимости, позволяющие нарушителю получать несанкционированный доступ к ресурсам компьютера, на котором установлено данное программное обеспечение.



# Аудит в Windows

В любой версии Windows есть система аудита, в которой есть возможность отслеживать и заносить в журнал данные о том, когда, где и при каких обстоятельствах, а еще при помощи какой именно программы произошли те или иные события, которые повлекли за собой удаление папки или позволили стереть или изменить важный файл. Но по умолчанию аудит не работает, так как для этого требуется задействовать определенную мощность системы. А нагрузка может быть слишком высокой, поэтому политики аудита ведут выборочную запись тех событий, которые по-настоящему важны.

## Журнал аудита

Для просмотра журнала аудита в Windows используется оснастка Event Viewer консоли администрирования, эту же оснастку можно использовать и для просмотра других системных журналов. Оснастка Event Viewer отображает события, регистрируемые системой при выполнении различных операций. Ее можно запустить путем ввода команды `eventvwr` в строку Run и нажав ОК.

По умолчанию события записываются в одном из трех журналов:

- **System (Система):** отображает системные события Windows.
- **Application (Приложения):** отображает события, записанные установленными приложениями.
- **Security (Безопасность):** отображает записи регистрации входа и выхода в систему, а также действия, связанные с доступом к файлам и папкам.

В каждом из них при помощи Event Viewer можно просмотреть, какие действия выполнялись в системе. Например, в журнале System записывается информация о запуске и остановке системных служб.

# Аудит в Windows

## Журнал аудита

Просматривать журнал аудита разрешено только пользователям, обладающим привилегией аудитора. Эти ограничения доступа действуют и в том случае, когда системный раздел жесткого диска отформатирован с использованием файловой системы, отличной от NTFS. Все пользователи, которые могут читать журнал аудита, могут и очищать его. Факт очистки журнала регистрируется сразу после очистки.

Размер журнала аудита ограничен, максимальный размер составляет по умолчанию от 512K в Windows NT 4 до 20M в Windows 7 и выше. Администратор операционной системы может менять это значение.

Также администратор может определить поведение операционной системы при переполнении журнала аудита. По умолчанию события перезаписываются по мере необходимости, начиная с самых старых. В политике безопасности может быть выставлена опция «аварийно завершать работу операционной системы при переполнении журнала аудита». В этом случае после перезагрузки операционной системы работать с ней сможет только администратор. Чтобы вернуть операционную систему в обычный многопользовательский режим, администратор должен очистить журнал аудита, сбросить в реестре соответствующий флаг и перезагрузить систему.

Добавлять записи в журнал аудита могут только субъекты доступа, обладающие соответствующей привилегией. По умолчанию эта привилегия предоставляется только псевдопользователю SYSTEM, менять данную установку не следует. Если предоставить данную привилегию какому-то физическому пользователю, он тем самым получит возможность записывать в журнал аудита произвольную информацию, в том числе и компрометирующую других пользователей.

# Аудит в Windows

## Категории событий

Множество событий, информация о которых записывается в журнал аудита, определяется политикой аудита, которую определяют пользователи-аудиторы. Windows позволяет регистрировать в журнале аудита события следующих категорий:

- вход/выход пользователя в/из системы;
- аутентификация пользователя<sup>[1]</sup>;
- доступ субъектов к локальным объектам;
- доступ субъектов к объектам активного каталога;
- использование субъектами доступа опасных привилегий;
- изменения в списке пользователей;
- изменения в политике безопасности;
- системные события;
- запуск и завершение процессов.

Для каждой категории событий могут регистрироваться либо только успешные события, либо только неуспешные, либо и те, и другие, либо никакие.

# Аудит в Windows

## Политика аудита и атрибуты защиты

По умолчанию в Windows, начиная с Windows Vista, реализуются следующие настройки политики аудита:

- на контроллерах доменов:
  - вход/выход пользователя в/из системы — только успешные попытки;
  - аутентификация пользователя — только успешные попытки;
  - доступ субъектов к объектам активного каталога — только успешные попытки;
  - изменения в списке пользователей — только успешные попытки;
  - изменения в политике безопасности — только успешные попытки;
  - системные события — только успешные попытки;
- на рабочих станциях:
  - вход/выход пользователя в/из системы — только успешные попытки;
  - аутентификация пользователя — только успешные попытки.

В доменах Windows политика аудита интегрирована в групповую политику и наследуется в соответствии с правилами наследования групповой политики.

# Аудит в Windows

## Политика аудита и атрибуты защиты

Порядок регистрации событий при доступе субъектов к объектам определяется не только политикой аудита, но и атрибутами защиты объекта. Как уже упоминалось выше, в состав дескриптора защиты объекта может входить системный список контроля доступа (SACL), определяющий порядок регистрации событий аудита при доступе субъектов к данному объекту. Так же, как и DACL, SACL представляет собой список переменной длины, элементами которого являются ACE, имеющие точно такой же формат, как и ACE, входящие в состав DACL.

В отличие от ACE из DACL, ACE в SACL всегда имеют тип «регистрирующий ACE» (system audit ACE). Для объектов активного каталога также поддерживается тип «объектно-специфичный регистрирующий ACE». Кроме того, для всех объектов поддерживается (хотя эта возможность недокументированна) system audit compound ACE, позволяющий отдельно описывать параметры регистрации доступа к объектам для каждой пары «субъект-клиент + субъект-сервер».

Элементы контроля доступа, входящий в SACL, могут иметь все флаги, которые может иметь ACE, входящий в DACL. Кроме того, ACE из SACL могут иметь еще два флага:

- **SUCCESSFUL\_ACCESS\_ACE\_FLAG** - если этот флаг установлен, будут регистрироваться в журнале аудита все успешные обращения к объекту субъекта, идентификатор которого записан в ACE, по любому из методов доступа, перечисленных в маске доступа ACE;
- **FAILED\_ACCESS\_ACE\_FLAG** - если этот флаг установлен, будут регистрироваться в журнале аудита все неуспешные обращения к объекту субъекта, идентификатор которого записан в ACE, по любому из методов доступа, перечисленных в маске доступа ACE.

# Аудит в Windows

## Политика аудита и атрибуты защиты

Если в ACE установлены оба флага, регистрируются любые обращения субъекта к объекту по перечисленным методам доступа, как успешные, так и неуспешные. Если в ACE установлен флаг INHERIT\_ONLY\_ACE, при доступе субъектов к объекту ACE игнорируется.

- Поскольку все ACE в SACL однотипны, порядок их взаимного расположения не имеет значения.
- Если в дескрипторе защиты объекта SACL отсутствует, обращения субъектов к этому объекту не регистрируются.
- При создании нового объекта SACL назначается объекту по тем же правилам, что и DACL.
- При создании нового объекта SACL назначается объекту по тем же правилам, что и DACL. При наследовании ACE флаги SUCCESSFUL\_ACCESS\_ACE\_FLAG и FAILED\_ACCESS\_ACE\_FLAG остаются неизменными.

# Аудит в Windows

## Политика аудита и атрибуты защиты

Для того чтобы событие, связанное с доступом субъекта к объекту, было зафиксировано в журнале аудита, необходимо одновременное выполнение следующих двух условий:

- Политика аудита операционной системы допускает регистрацию в журнале аудита событий, связанных с успешным (или, соответственно, неуспешным) доступом субъектов к объектам.
- SACL объекта содержит хотя бы один ACE, в котором:
  - идентификатор субъекта относится к субъекту, открывающему объект;
  - установлен флаг `SUCCESSFUL_ACCESS_ACE_FLAG` (или, соответственно, `FAILED_ACCESS_ACE_FLAG`) и не установлен флаг `INHERIT_ONLY_ACE`;
  - после отображения отображаемых прав доступа пересечение маски доступа ACE и маски доступа, содержащей права, запрашиваемые субъектом, не пусто.

Таким образом, глобальные настройки политики аудита в отношении доступа субъектов к объектам играют роль фильтра, позволяющего временно запретить регистрацию успешных или неуспешных попыток доступа всех субъектов ко всем объектам операционной системы.

# Аудит в Windows

## События и задачи

Начиная с Windows Vista, администратор может привязать к каждому типу событий аудита одну или несколько задач следующего вида:

- вывести текстовое сообщение в текущую терминальную сессию;
- отправить электронное письмо на заданный адрес;
- запустить заданную программу.

К сожалению, во всех трех случаях подробные сведения о зарегистрированном событии не передаются задаче в качестве параметров. Фактически, данный механизм позволяет интерактивно оповещать администратора безопасности только о факте наступления того или иного события (например, неудачной попытке входа пользователя в систему), но подробности события (с какой учетной записью и каким сетевым адресом оно связан) администратор должен будет выяснять самостоятельно, лично просмотрев журнал аудита.



# Аудит в Windows

## Подписки на события аудита

Начиная с Windows Vista, в подсистеме аудита Windows поддерживается механизм, позволяющий автоматически перенаправлять записи о событиях определенных видов с одних компьютеров на другие.

Перенаправление записей аудита реализуется посредством так называемых *подписок*, создаваемых с помощью оснастки Event Viewer консоли администрирования. Для управления подписками необходимо иметь полномочия администратора как в системе-источнике, так и в системе-сборщике. Кроме того, может потребоваться дополнительная настройка пакетного фильтра.

Средства автоматического обнаружения вторжений в современных версиях Windows отсутствуют. Microsoft ISA Server, согласно документации, включает в себя встроенную систему обнаружения вторжений, но она настолько примитивна, что вряд ли ее можно всерьез относить к данному классу средств защиты информации.



# Аудит в UNIX

В современных операционных системах семейства UNIX реализуются два принципиально различных подхода к организации аудита.

Первый из них основан на применении традиционных для UNIX демонов регистрации событий `syslogd` и `klogd`, изначально разработанных не столько для поддержания безопасности операционной системы, сколько для выявления и устранения ошибок конфигурирования, сетевых неполадок, взаимных несовместимостей пакетов программного обеспечения и т. п.

Второй, альтернативный подход появился сравнительно недавно, он основан на переносе в среду UNIX архитектуры и интерфейсов подсистемы аудита, изначально свойственных операционным системам семейства Windows. Если UNIX-система включена в состав гетерогенной сети, большинство узлов которой работают под управлением Windows, единообразное построение подсистем аудита всех операционных систем становится серьезным преимуществом, поскольку позволяет, например, распространять политики аудита на большие подмножества узлов сети независимо от того, под управлением какой операционной системы работает каждый конкретный узел того или иного подмножества.

В состав большинства современных UNIX-систем могут включаться обе подсистемы аудита, при этом они могут работать как независимо одна от другой, так и во взаимодействии.

# Аудит в UNIX

## syslogd, klogd

Вначале мы рассмотрим первый, более традиционный и более часто применяемый подход к организации аудита в UNIX, основанный на использовании демонов `syslogd` и `klogd`.

Регистрация событий, связанных с безопасностью операционной системы, не отделяется этими демонами от регистрации других системных событий, поэтому при описании данного подхода мы будем использовать термин «аудит» расширенно — как процедуру регистрации любых событий, имевших место в ходе функционирования операционной системы и не обязательно непосредственно связанных с ее безопасностью.

Демон `syslogd` предоставляет услуги прикладным и системным программам, демон `klogd` реализует аудит операций, выполняемых ядром операционной системы. Основным назначением этих демонов является поддержка единой в масштабах операционной системы политики аудита, предписывающей выполнение определенных действий для каждого типа регистрируемых событий.

Каждый раз, когда прикладная или системная программы выполняет потенциально аудируемое действие, информация об этом действии передается демону `syslogd`, тот сверяется с текущей политикой аудита и принимает решение о порядке регистрации данного события.

Получение информации демоном `syslogd` обычно реализуется через сокет `/dev/log` (для локальных клиентов) или UDP-порт 514 (для удаленных клиентов).

# Аудит в UNIX

## syslogd, klogd

Демон klogd получает информацию через специальный файл `/proc/kmsg` или системный вызов ядра `sys_syslog`. Идентификаторы процессов демонов аудита хранятся в десятичном виде в текстовых файлах `/var/run/syslogd.pid` и `/var/run/klogd.pid` соответственно.

Действующая политика аудита описывается текстовым файлом `/etc/syslog.conf`. Каждая строка этого файла описывает одно элементарное правило политики аудита и включает в себя два основных поля:

- *селектор* — описывает условия применимости данного правила;
- *действие* — описывает действие, которое должно быть выполнено в результате применения данного правила.

Поле *селектора* в общем случае имеет следующий вид:

**<источник>. <модификатор> <приоритет>.**

Подполе <источник> описывает подсистему операционной системы, которая обращается к демону `syslogd` с намерением зарегистрировать то или иное событие (например, `ftp`, `kern`, `mail` и т.д.). В одном селекторе можно указывать несколько источников через запятую.

Подполе <модификатор> описывает реакцию на событие в зависимости от его события.

Подполе <приоритет> определяет приоритет события (принимает нечисловое значение, например, `alert`, `crit` и т.д.).

В одной строке файла `syslogd.conf` можно описывать несколько селекторов, они разделяются точкой с запятой (;).

# Аудит в UNIX

## syslogd, klogd

Поле *действия* описывает конкретное действие, которое должен предпринять демон `syslogd` при регистрации события аудита, соответствующего указанному селектору. Поддерживаются следующие типы действий:

- запись информации о событии в файл — указывается имя файла. Обычно сразу после записи информации о событии выполняется принудительный сброс дискового кэша для данного файла, этим гарантируется сохранение информации в случае внезапного аварийного завершения работы операционной системы. Принудительный сброс кэша можно отменить для любого конкретного файла, указав перед именем файла символ - (минус);
- выдача информации о событии на терминал или консоль — указывается имя файла в директории `/dev`, соответствующее данному терминалу и консоли;
- запись информации о событии в указанный именованный канал — указывается имя канала, которому предшествует символ «|»;
- перенаправление информации о событии демону `syslogd` другого компьютера (514 порт UDP) — указывается имя компьютера, которому предшествует символ «@». При неаккуратном описании порядка перенаправления сообщений аудита с одних компьютеров на другие могут возникать циклы, что может приводить к перегрузке сети и переполнению файлов аудита;
- интерактивное оповещение о событии одного или нескольких пользователей — указываются имена пользователя через запятую;
- интерактивное оповещение о событии всех пользователей, работающих в данный момент с операционной системой — указывается одиночный символ «\*».

# Аудит в UNIX

## syslogd, klogd

Рассмотрим фрагмент файла `syslog.conf` как пример описания политик аудита.

```
# Все критические сообщения, кроме исходящих от ядра,  
# записывать в файл /var/adm/critical  
*.crit;kern.none /var/adm/critical  
# Все сообщения от ядра записывать в файл /var/adm/kernel  
kern.* /var/adm/kernel  
# Критические сообщения от ядра также выводить на консоль и  
# перенаправлять на компьютер netaud  
kern.crit /dev/console  
kern.crit @netaud  
# Не очень важные сообщения от ядра записывать в файл  
#/var/adm/kernel-info  
kern.!err /var/adm/kernel-info  
# Информационные сообщения от почтового сервера выводить на  
# двенадцатый терминал  
mail.=info /dev/tty12  
# Все остальные сообщения от почтового сервера записывать в файл  
#/var/adm/mail  
mail.*;mail.!=info /var/adm/mail  
# Информационные сообщения от почтового сервера и ШТР-сервера  
# записывать в файл /var/adm/info  
mail,news.=info /var/adm/info
```

# Аудит в UNIX

## syslogd, klogd

При обновлении файла `syslog.conf` новая политика аудита вступает в силу только после перезагрузки демона `syslogd` или после того, как этот демон получит сигнал `SIGHUP`.

Прикладные программы UNIX интерпретируют данный сигнал как обрыв связи удаленного клиента с терминалом и обычно, получив его, аварийно завершают работу, но демон `syslogd` воспринимает сигнал `SIGHUP` иначе — как требование обновить политику аудита.

В большинстве UNIX-систем большая часть сообщений аудита традиционно записывается в файл `/var/log/messages` или `/var/adm/ log/messages`. Для хранения сообщений о попытках аутентификации обычно используется файл `/var/log/auth.log`.

Журналы аудита UNIX представляют собой обычные текстовые файлы. Их просмотр и анализ может проводиться как обычными утилитами просмотра текстов, так и более продвинутыми программными средствами, облегчающими работу с аудитом.

# Аудит в UNIX

## syslogd, klogd

В отличие от Windows, в UNIX нет встроенных средств, позволяющих жестко ограничивать максимальные размеры журналов аудита. Неадекватная политика аудита в UNIX потенциально может приводить к исчерпанию свободного места на жестких дисках компьютера. Для предотвращения данной угрозы могут применяться следующие меры:

- ограничение доступа к 514 порту UDP пакетным фильтром;
- размещение файлов аудита на отдельном разделе жесткого диска;
- запуск демона `syslogd` от имени псевдопользователя с ограниченными полномочиями.

Удаление из файлов аудита устаревшей информации традиционно реализуется путем регулярного запуска демоном `cron` специальной утилиты `logrotate`, конфигурация которой описывается файлом `/etc/logrotate.conf`.

Серьезным недостатком подсистемы аудита UNIX является то, что доступ к демону `syslogd` в общем случае предоставляется любым программам, как системным, так и прикладным. При этом клиентская программа не только передает демону текст сообщения, но и самостоятельно указывает источник и приоритет регистрируемого события. Более того, существует специальная утилита командной строки `logger`, позволяющая обычному непривилегированному пользователю передать демону `syslogd` произвольную информацию от имени произвольной подсистемы операционной системы и присвоить этой информации произвольный уровень значимости.



# Аудит в UNIX

## auditd

Механизм регистрации событий, реализуемый демонами `syslogd` и `klogd`, в основном предназначен не столько для поддержания безопасности системы, сколько для выявления программных и аппаратных неисправностей, тестирования и отладки новых компонент операционной системы и т. п.

Применять данный механизм для регистрации событий, связанных с безопасностью системы, не очень удобно, и в некоторых операционных системах, принадлежащих к семейству UNIX, для этой задачи предназначен отдельный, дополнительный демон аудита, которому чаще всего назначается имя `auditd`.

Концептуально реализации данного демона в разных UNIX-системах обычно очень похожи на реализацию подсистемы аудита в Windows. Однако технические детали разных реализаций могут очень сильно отличаться одна от другой. Рассмотрим пару примеров.

**В операционной системе AIX** демон `auditd` предназначается главным образом для регистрации обращений пользователей к файлам. Конфигурация демона описывается в файлах `config` и `objects`, размещаемых в директории `/etc/security/audit`. Данные аудита записываются либо в псевдофайл `/audit/trail`, либо на логическое устройство `/dev/audit`.

# Аудит в UNIX

## auditd

Для считывания данных аудита уполномоченным пользователем используются специальные утилиты `auditptr` и `auditstream`. Формат данных аудита при этом выглядит примерно следующим образом:

event	login	status	time	command
-----	-----	-----	-----	-----
S_NOTAUTH_READ	root	OK	Thu Nov 1 14:07:05 2012	cat
S_NOTAUTH_READ	root	OK	Thu Nov 1 14:07:05 2012	cat
FILE_Unlink	root	OK	Thu Nov 1 14:07:09 2012	vi
S_NOTAUTH_READ	root	OK	Thu Nov 1 14:07:09 2012	vi
S_NOTAUTH_READ	root	OK	Thu Nov 1 14:07:09 2012	vi
S_NOTAUTH_READ	root	OK	Thu Nov 1 14:07:09 2012	vi
S_NOTAUTH_WRITE	root	OK	Thu Nov 1 14:07:13 2012	vi
FILE_Unlink	root	OK	Thu Nov 1 14:07:13 2012	vi
FILE_Unlink	root	OK	Thu Nov 1 14:07:20 2012	vi
S_NOTAUTH_READ	ash	OK	Thu Nov 1 14:09:39 2012	cat
S_NOTAUTH_READ	ash	OK	Thu Nov 1 14:09:39 2012	cat

# Аудит в UNIX

## auditd

В **Mac OS X** демон `audit` записывает регистрируемые данные в один или несколько файлов, обычно расположенных в директории `/var/audit`. Политика аудита для данного демона описывается конфигурационными файлами `audit-class`, `audit-event`, `audit-control` и `audit-user`. Для каждого пользователя операционной системы порядок регистрации событий, связанных с деятельностью этого пользователя, задается индивидуально, кроме того, определен порядок регистрации событий по умолчанию.

Механизм регистрации событий, реализуемый демоном `audit` в **Mac OS**, во многом похож на реализацию аудита в **Windows**. Аналогично категориям событий аудита в **Windows**, в **Mac OS** события аудита объединяются в так называемые классы, при этом порядок регистрации событий отдельно определяется для успешных и неуспешных событий каждого класса. Для каждого пользователя определяются так называемые флаги аудита — битовая маска, в которой каждый бит соответствует одному классу потенциально регистрируемых событий. По умолчанию в **Mac OS** задано 19 классов событий аудита, при этом система их классификации запутана и не слишком удобна для практического использования.

Для некоторых событий возможно интерактивное оповещение администратора с помощью автоматического запуска специального скрипта `audit.warn`, который в качестве параметра получает текстовую строку, содержащую описание зарегистрированного события. По умолчанию скрипт `audit.warn` просто записывает текущее время и переданный ему текст в конец файла `/etc/security/audit.messages`, но администратор операционной системы может менять код скрипта произвольным образом.